Amendments to the Specification

On page 1, before the section entitled "BACKGROUND OF THE INVENTION" insert the following text:

<p style="text-align:center">RELATED APPLICATIONS</p>

This application claims the priority of U.S. Provisional Application Serial No. 60/206,571, filed May 23, 2000.

On page 3, the 3rd paragraph beginning on line 10:

Figure 1 presents the format for a first gene expression database relating to expression results, the ~~GeneExpress~~ GeneExpress® expression result format.

On page 3, the 5th paragraph beginning on line 15:

Figure 3 presents the format for a first gene expression database relating to samples, the ~~GeneExpress~~ GeneExpress® sample format.

On page 3, the 7th paragraph beginning on line 19:

Figure 5 presents the format for a first gene expression database relating to genes, the ~~GeneExpress~~ GeneExpress® gene format.

On page 4, the 5th paragraph beginning on line 16:

Over time, the users of the ~~gene express~~ GeneExpress® database at a particular site will develop a large number of datasets, and will need a mechanism for organizing them so that related data are grouped together. Some data will be accessible to all users, some only to selected users, and some only to the user who created it. There is also a need for predefined read-only gene sets, selected according to commonly used criteria (for example, "all genes involved in the oxidative phosphorylation pathway", or "all genes on rat chromosome 2"); these are preferably publicly available to all users.

On page 6, the 3rd paragraph beginning on line 11:

With no integration, there is no change ~~toany~~ <u>to any</u> of the multiple databases. However, the query and analysis tools run would have to run separately on each of ~~th~~ <u>the</u> multiple databases. In such circumstances, gene matching can be done only by means of sequence homology comparisons. Thus, one cannot analyze data from both databases together.

On page 7, the 2nd paragraph beginning on line 9:

Figure 1 presents the format for a first gene expression database relating to expression results, the ~~GeneExpress~~ <u>GeneExpress</u>® expression result format.

On page 7, the 4th paragraph beginning on line 14:

Figure 3 presents the format for a first gene expression database relating to samples, the ~~GeneExpress~~ <u>GeneExpress</u>® sample format.

On page 7, the 6th paragraph beginning on line 18:

Figure 5 presents the format for a first gene expression database relating to genes, the ~~GeneExpress~~ <u>GeneExpress</u>® gene format.

On page 8, the 1st paragraph beginning on line 3:

In achieving this overall integration, the query and analysis ~~tolls~~ <u>tools</u> are preferably extended.

On page 8, the 3rd paragraph beginning on line 8:

Another preferred aspect to extending the gene index is to develop new utilities for loading the existing data (and new data) into the developed new classes. Such an approach can allow for the ~~preservatoion~~ <u>preservation</u> of custom associations in ~~thew~~ <u>the</u> database <u>that were</u> previously developed.

On page 8, the 5th and 6th paragraphs beginning on line 17:

In loading the existing gene data into new classes, there are preferably several data tools which are employed, ~~inveluding~~ <u>including</u> gene fragment classification and gene fragment annotation.

Gene fragment classification sorts gene fragments into two classes: (1) Known (with a GenBank accession number) and (2) Not known (not in GenBank). The ~~Knwn~~ <u>Known</u> classification employs name-based or sequence-based matching methods. The Not known classification utilizes sequence classification workbench to provide support for clustering with regard to a predetermined set of sequences; classification as novel/known/unknown sequence; association with public sequence clusters (Unigene, STACK); and periodic reclassification.

On page 9, the 2nd paragraph beginning on line 7:

Preferably, there are a number of methods employed in gene ~~fragmernt~~ <u>fragment</u> classification. These include name based matching, sequence based matching, and manual data curation.

On page 9, in the 5th paragraph beginning at line 18:

~~Regardng~~ <u>Regarding</u> manual data curation, gene fragments without a Unigene match are reviewed to detect potential sequence data contamination.

On page 10, the 1st paragraph beginning at line 1:

The ~~gene express~~ <u>GeneExpress</u>® database explorer interface (henceforth, "GX Explorer™") preferably consists of a main Workspace window and one or more query and analysis tool windows. The Workspace window provides project management functionality, and also acts as a launcher for the various tool windows. Tool windows are of several types, including, but not limited to, Sample Set, Gene Set, Gene Signature, E-Northern™, Fold Change, Gene Signature Differential, Expression Data, and Cluster Analysis. Tool windows are preferably not displayed until the user explicitly launches them. A user can preferably display as many windows of each type as he/she wishes, subject to the memory limitations of his/her client PC.

On page 12, the 5th paragraph beginning at line 17:

Preferably, a user must have a username and password to access the ~~Gene Express~~ GeneExpress® system. User accounts are created by an administrator using the interface described below; the administrator assigns the user an initial password, which the user can then change.

On page 13, the 1st – 5th paragraphs beginning at line 3:

There is one special user that is predefined when ~~Gene Express~~ GeneExpress® is installed, called "admin". A user must know the admin password in order to use the administrative UI functions, such as creating user accounts. The admin user belongs to a predefined group, also called "admin".

To access ~~Gene Express~~ GeneExpress®, a user runs the ~~Gene Express~~ GeneExpress® Explorer™ application (hereafter called GX Explorer™), typically by double-clicking a desktop icon from Windows®. GX Explorer™ first displays a login dialog, in which the user enters his/her username and password. The dialog provides a Help button, through which the user can obtain information such as how to obtain a ~~Gene Express~~ GeneExpress® user account, the client system requirements, and how to log in.

When the user enters a valid username and password, the dialog is hidden and the main GX Explorer™ window is displayed.

The user administration operations are invoked through a separate ~~Gene Express~~ GeneExpress® Admin application. A user can also change information in his/her own profile through the "Change Password" and "User Profile" commands in GX Explorer™.

To start the ~~Gene Express~~ GeneExpress® Admin application, the administrator runs ~~Gene Express~~ GeneExpress® Admin from the Windows Start menu. A login dialog is displayed; the administrator enters the admin username and password. The main ~~GX~~ GX™ Admin window is then displayed.

On page 15, the last paragraph beginning on line 18:

The Workspace window contains a tree view, called the Workspace Navigator, used to display and manage the data objects maintained by ~~Gene Express~~ GeneExpress®. Data objects are displayed within project folders, represented by folder icons; the icons used to represent the

data objects themselves vary and are used to indicate the object type. The look and feel is similar to that of the ~~Macintosh~~ Macintosh® Finder or the ~~Windows~~ Windows® Explorer; it resembles the former more than the latter in that folders and objects are displayed within the same tree view, so that the objects in multiple folders can be viewed simultaneously.

On page 16, the 2nd paragraph beginning at line 7:

The tree view display is coordinated with the tabular display of folder and object attributes shown in the right hand pane of the GX™ Workspace window. Users can expand and collapse folders with the usual UI gestures (clicking the plus and minus icons or double-clicking the folder icons); the tabular display is preferably refreshed when this happens so that the attributes line up with the objects and folders to which they apply ~~to~~.

On page 17, the last paragraph beginning at line 22 and continuing to page 18:

As an aspect of project management, ~~Gene Express~~ GeneExpress® enables users to organize data objects into project folders, which are represented using folder icons in the Navigator. Each user has his/her own top-level folder, with the same name as his/her Workspace Manager username. It may be subdivided into project-specific folders. There is a predefined user named "Public", whose folders contain various read-only data objects predefined by ~~Gene Express~~ GeneExpress®. These read-only objects may be copied into a user's own folder, and the copies may be edited. Folders are assigned permissions, so that other users or groups of users may be granted access; by default folders are not accessible by anyone other than the user who created them.

On page 22, the 1st paragraph beginning on line 3:

A Gene Set represents a list of gene fragments, each on a specific Affy chip type, selected according to user-supplied criteria. A user defines a Gene Set by performing one of various types of queries against the ~~Gene Express~~ GeneExpress® Fragment Index (a.k.a. the GX™ Index). The internal representation of a Gene Set is a list of gene IDs, which are combinations of the numeric Affy item_ids for each fragment with the two digit IDs of the Affy chip containing each fragment instance. The chip IDs are required to distinguish between cases

where the same Affy gene fragment is tiled on more than one chip type; there are several hundred such cases.

On page 22, the 3rd paragraph beginning at line 13:

A Gene Set is initially defined by the results of a query against the GX™ Index at a particular time. The data in the GX™ Index change over time; not only are new chip types released by Affymetrix and used to generate expression data for ~~Gene Express~~ GeneExpress®, but also new information is generated linking known genes and EST clusters to existing Affy gene fragments. Thus, the same query run at a later time may return a different list of gene IDs. It is important to remember though that these changes in the data content of the GX™ Index will not affect the list of gene IDs in existing Gene Sets.

On page 23, the 2nd paragraph beginning at line 8:

The Workspace Manager supports other types of data objects which are associated with the various analyses that can be run in ~~Gene Express~~ GeneExpress®: Gene Signatures, Gene Signature Differentials, Fold Change Analyses, Electronic Northerns, and Cluster Analyses. The user can perform all the standard workspace operations on these objects (copy, move, rename, etc.), or can load the object for visualization or further analysis. Although there may be situations where the actual data results are stored with these objects, usually the Workspace Manager only stores the information necessary to rerun the analysis (e.g., the Sample Set and thresholds for a Gene Signature).

On page 24, the 4th paragraph beginning on line 12:

The Open operation loads one selected object for visualization and (if appropriate) for editing. This operation is highly polymorphic and will display different information for different data object types. Generally, when the user selects this operation, GX Explorer™ will create a new window of the appropriate type (a Sample Set window for a Sample Set, a Gene Signature window for a Gene Signature, etc.) and open it to a summary view of the object. The contents of the summary view will vary according to the object type, but will usually be the first view displayed after a query or analysis is run.

On page 29, the 1st and 2nd paragraphs beginning at line 1:

The Sample Set window has an additional pull-down menu, Species, which is not present in the Gene Search window. The pull-down menu requires the user to choose between the possible species values in the Sample DB. The value set in the species menu is saved in the user's preferences. Changing the selection in the species menu preferably resets the Search Parameters tree and ~~clear~~ clears the Conditions list. Note that the choice of the species preferably affects whether certain classes and attributes are present in the Search Parameters tree. For example the [Donor Animal] class preferably is not present if the species is Human, and the [Donor Human] class is not present if the species is rat or mouse. However, the [Donor Cell-Line] class may be present for any species, together with the other appropriate Donor subclass, since cell line and tissue samples may be mixed in a Sample Set. The search interface code preferably detects ANDs of constraints on attributes from different Donor subclasses and prevent the user from executing the query if they don't make sense.

Sample searches are constrained by another implicit condition, which is the ownership of the sample data. An instance of the ~~Gene Express~~ GeneExpress® data warehouse may contain private data from samples provided by several external alliance partners. One alliance partner preferably is not permitted to see another partner's private data. Since all analysis operations and data exports require the user to specify a SampleSet, external users are restricted from having access to another alliance's data by only returning samples in a Sample Search that are either public or owned by the alliance partner.

On page 35, the 1st paragraph beginning at line 1:

Note that, if an attribute or path is multi-valued, then, in order to satisfy a condition, it is only necessary for one value of the attribute or path to be in the relationship with the field value. For example, it a user specified the condition Comments matches "%~~foobar~~ foot%", then this condition would be satisfied by any object with a Comments attribute containing at least one string containing the sub-string "~~foobar~~ foot". It would not require an object to have "~~foobar~~ foot" as a sub-string of all its Comment values.

On page 37, in the last paragraph beginning on line 18:

The user can select one or more objects and perform an operation on them, such as saving them as a sample or gene set. Objects can be selected individually by clicking on their rows, or en masse by dragging over multiple rows; multiple ~~discontiguous~~ non-contiguous rows can be selected by control-clicking.

On page 39, the 2nd paragraph beginning on line 6:

There are two kinds of form views: views generated automatically based on the OPM metadata for the associated class and views that are defined by configuration files, which are stored on the ~~Gene Express~~ GeneExpress$^{®}$ application server. The configuration file specifies the attributes that are displayed, the types of fields used to represent them, and the layout of the fields. Configuration files are created and modified using a combination of a visual form editor tool and manual text editing.

On page 49, the 8th paragraph beginning on line 19:

The History operation prints the "genealogical" information for the current Gene Set. Information displayed for Gene Sets including, but not limited to, gene set name and description, query conditions, query date, versions of GX™ Index (schema and build), and log of operations performed, including set operations and manual editing.

On page 50, the 1st paragraph beginning on line 1:

If the Gene Set was exported from an analysis result, instead of displaying the query data, display the genealogy information from the analysis object.

On page 50, the 4th paragraph beginning on line 19:

For example, suppose the ~~Gene Express~~ GeneExpress$^{®}$ data warehouse contained the present/absent/marginal/unknown call values shown in the table below, for the sample set S = {s1, s2, s3, s4} and the genes {g1, g2, g3, g4, g5, g6, g7, g8, g9}. (In reality there would be data for thousands of genes, but only nine genes are shown for illustration.) At the bottom of the column for each gene are shown the percentages computed from the numbers of present, absent and marginal calls for each gene across sample set S.

On page 81, the last paragraph beginning on line 22:

With regard to the Implementation Overview, the ~~Gene Express~~ GeneExpress® user interface is built in Java, using CORBA to communicate with a Runtime Engine and with the sample, GX Index, and workspace management servers. The Runtime Engine will handle all compute- and memory-intensive operations, such as clustering, computation of Gene Signatures, and fold change analysis.